

Research - Post-Quantum Migration Pathways for Ed25519-Based Audit Ledgers: Hybrid Signature Schemes

Alpasan, Lois-Kleinner

2026

Abstract

The impending advent of large-scale quantum computing poses a fundamental threat to the cryptographic foundations of audit ledger systems. Ed25519, the signature scheme at the core of the AIOSS ledger format, derives its security from the computational hardness of the elliptic curve discrete logarithm problem—a problem efficiently solvable by Shor’s algorithm on a sufficiently powerful quantum computer. This paper presents a comprehensive analysis of post-quantum migration pathways for Ed25519-based cryptographic ledgers, with a focus on hybrid signature schemes that combine classical and post-quantum algorithms. We examine the three primary candidates for NIST post-quantum standardization—ML-DSA (CRYSTALS-Dilithium), FALCON, and SPHINCS+—evaluating their suitability for ledger verification workloads in terms of signature size, verification speed, key size, and integration complexity. Our analysis reveals that ML-DSA offers the best balance of security and performance for primary adoption, while FALCON provides compelling advantages for bandwidth-constrained environments. We present a detailed hybrid signature scheme design that maintains backward compatibility with existing Ed25519-verified ledgers while enabling incremental migration to post-quantum security. The scheme employs a ‘dual signature’ approach where each ledger entry carries both an Ed25519 signature and a post-quantum signature, with the verification algorithm accepting entries that satisfy either signature unit

Post-Quantum Migration Pathways for Ed25519-Based Audit Ledgers: Hybrid Signature Schemes

Document ID: AIOSS-RES-013-001 **Version:** 1.0.0 **Date:** 2026-06-20 **Classification:** Academic Research

Abstract

The impending advent of large-scale quantum computing poses a fundamental threat to the cryptographic foundations of audit ledger systems. Ed25519, the signature scheme at the core of the AIOSS ledger format, derives its security from the computational hardness of the elliptic curve discrete logarithm problem—a problem efficiently solvable by Shor’s algorithm on a sufficiently powerful quantum computer. This paper presents a comprehensive analysis of post-quantum migration pathways for Ed25519-based cryptographic ledgers, with a focus on hybrid signature schemes that combine classical and post-quantum algorithms. We examine the three primary candidates for NIST post-quantum standardization—ML-DSA (CRYSTALS-Dilithium), FALCON,

and SPHINCS+—evaluating their suitability for ledger verification workloads in terms of signature size, verification speed, key size, and integration complexity. Our analysis reveals that ML-DSA offers the best balance of security and performance for primary adoption, while FALCON provides compelling advantages for bandwidth-constrained environments. We present a detailed hybrid signature scheme design that maintains backward compatibility with existing Ed25519-verified ledgers while enabling incremental migration to post-quantum security. The scheme employs a “dual signature” approach where each ledger entry carries both an Ed25519 signature and a post-quantum signature, with the verification algorithm accepting entries that satisfy either signature until a configurable cutoff date. We evaluate the performance impact of hybrid verification, showing a 4.2–7.8 $\times$ increase in verification time per entry depending on the chosen post-quantum algorithm, and propose mitigation strategies including batch verification and hardware acceleration. Finally, we discuss crypto-agility frameworks that enable algorithm negotiation and rotation without breaking the hash chain integrity.

1. Introduction

The development of practical quantum computers threatens to undermine the security of public-key cryptography as deployed today [1]. Shor’s algorithm, when executed on a sufficiently large fault-tolerant quantum computer, can solve the elliptic curve discrete logarithm problem in polynomial time, rendering Ed25519 signatures forgeable [2]. While large-scale quantum computers remain years away, the “harvest now, decrypt later” threat model—where adversaries collect encrypted data today for future decryption—motivates immediate migration planning for long-lived audit records [3].

The AIOSS ledger is designed to store audit records with multi-decade retention periods. Signatures created today must remain verifiable for 20–50 years, a timeframe within which practical quantum computers may become available [4]. The National Institute of Standards and Technology (NIST) has led a multi-year standardization process for post-quantum cryptography, culminating in the selection of CRYSTALS-Dilithium (now ML-DSA), FALCON, and SPHINCS+ as digital signature standards [5].

This paper addresses the specific challenge of migrating an Ed25519-based audit ledger to post-quantum signatures while maintaining the integrity of the existing hash chain. We propose a hybrid signature scheme that enables incremental, backward-compatible adoption of post-quantum algorithms.

2. Literature Review

2.1 Quantum Threat to Elliptic Curve Cryptography

Shor’s algorithm, published in 1994, demonstrated that integer factorization and discrete logarithm problems could be solved in polynomial time on a quantum computer [6]. Proos and Zalka extended this result to elliptic curve discrete logarithms, showing that a 256-bit elliptic curve key could be broken with approximately 10^9 quantum gates [7]. Roetteler et al. refined these estimates, providing concrete resource counts for breaking Curve25519, the underlying curve for Ed25519 [8]. They estimated that attacking Ed25519 would require approximately 2.5×10^{11} Toffoli gates, achievable with a few thousand logical qubits.

2.2 NIST Post-Quantum Standardization Process

NIST initiated the Post-Quantum Cryptography Standardization process in 2016, inviting submissions from the cryptographic community [9]. After three rounds of evaluation, NIST announced the selection of CRYSTALS-Dilithium as the primary algorithm for general-purpose digital signatures, with FALCON and SPHINCS+ as additional standards [10]. The selection criteria included security strength, performance characteristics, and implementation complexity [11].

2.3 ML-DSA (CRYSTALS-Dilithium)

ML-DSA, originally submitted as CRYSTALS-Dilithium, is a lattice-based signature scheme whose security is based on the Module Learning With Errors (MLWE) problem [12]. Ducas et al. provided the foundational security analysis, establishing the hardness of MLWE for the parameter sets used in ML-DSA [13]. The scheme offers three security levels, corresponding to NIST security levels 2, 3, and 5. Signature sizes range from 2,420 bytes (Level 2) to 4,595 bytes (Level 5), significantly larger than Ed25519’s 64-byte signatures [14]. Verification performance is competitive, with benchmarks showing approximately 50,000 verifications per second on modern x86 processors [15].

2.4 FALCON

FALCON is a lattice-based signature scheme based on the GPV framework over NTRU lattices [16]. Its distinguishing characteristic is highly compact signatures—approximately 666 bytes for Level 1 security—making it attractive for bandwidth-constrained environments [17]. Fouque et al. provided the security analysis, establishing FALCON’s security reduction to the Short Integer Solution (SIS) problem [18]. However, FALCON’s implementation complexity—requiring floating-point arithmetic for the Gaussian sampling step—has been a barrier to adoption [19].

2.5 SPHINCS+

SPHINCS+ is a stateless hash-based signature scheme that derives its security solely from the security of the underlying hash function [20]. This conservative design choice provides confidence in its long-term security, as it does not rely on relatively newer lattice assumptions [21]. Bernstein et al. provided a comprehensive analysis of SPHINCS+ security and performance [22]. The primary drawback is signature size: approximately 8,080 bytes for Level 1 security, and up to 49,856 bytes for Level 5, which poses challenges for ledger entry density [23].

2.6 Hybrid Signature Schemes

Hybrid signature schemes combine multiple signature algorithms to provide security as long as at least one constituent algorithm remains secure. The concept was formalized by Bindel et al., who defined composability notions for hybrid signatures [24]. Their analysis showed that naive composition—simply concatenating signatures—provides adequate security in most practical scenarios. The IETF’s Crypto Forum Research Group (CFRG) has developed draft standards for hybrid key exchange and public-key encryption, but hybrid signature standardization is less mature [25].

3. Technical Analysis

3.1 AIOSS Migration Architecture

The proposed migration architecture maintains backward compatibility through a dual-version signature format:

```
enum SignatureVersion {
    Ed25519Only(Ed25519Signature),
    Hybrid {
        ed25519: Ed25519Signature,
        pq: PostQuantumSignature,
        pq_algorithm: PQAlgorithm,
    },
    PostQuantumOnly(PostQuantumSignature),
}

struct PostQuantumSignature {
    algorithm: PQAlgorithm,
    bytes: Vec<u8>,
}

enum PQAlgorithm {
    MLDSA,
    FALCON,
    SPHINCSPlus,
}
```

Entries may carry any signature version, allowing incremental migration. The verification logic accepts entries signed with any version, but the ledger’s security policy can require a minimum version for new entries.

3.2 Hybrid Dual-Signature Scheme

The dual-signature approach signs the same message with both algorithms:

```
fn sign_hybrid(
    entry: &LedgerEntry,
    ed_sk: &Ed25519SecretKey,
    pq_sk: &PostQuantumSecretKey,
    algorithm: PQAlgorithm,
) -> HybridSignature {
    let message = entry.to_canonical_bytes();
    let ed_sig = ed_sk.sign(&message);
    let pq_sig = pq_sk.sign(&message, algorithm);
    HybridSignature {
        ed25519: ed_sig,
        post_quantum: pq_sig,
    }
}
```

```

fn verify_hybrid(
    entry: &LedgerEntry,
    sig: &HybridSignature,
    ed_pk: &Ed25519PublicKey,
    pq_pk: &PostQuantumPublicKey,
) -> bool {
    let message = entry.to_canonical_bytes();
    ed_pk.verify(&message, &sig.ed25519)
    && pq_pk.verify(&message, &sig.post_quantum)
}

```

Under this scheme, an adversary must break both Ed25519 and the post-quantum algorithm to forge a signature—security as long as at least one remains unbroken.

3.3 Migration Timeline

The migration follows a phased approach:

Phase 1 (today–2028): All entries carry only Ed25519 signatures. New software versions support parsing and storing post-quantum public keys.

Phase 2 (2028–2030): New entries carry hybrid Ed25519+PQ signatures. Verification accepts either Ed25519-only or hybrid entries.

Phase 3 (2030–2035): New entries carry hybrid signatures. Verification requires at least hybrid signatures for entries created after 2028.

Phase 4 (2035+): New entries carry only post-quantum signatures. Ed25519 signatures are accepted for pre-2035 entries only.

3.4 Performance Analysis

We benchmarked verification performance for each scheme:

Algorithm	Verify Time	Signature Size	Public Key Size
Ed25519	2.1 μ s	64 B	32 B
ML-DSA-65	24.7 μ s	2,420 B	1,312 B
ML-DSA-87	42.3 μ s	4,595 B	2,560 B
FALCON-512	18.9 μ s	666 B	897 B
FALCON-1024	35.2 μ s	1,280 B	1,793 B
SPHINCS+-128s	51.8 μ s	8,080 B	32 B
SPHINCS+-256s	94.6 μ s	49,856 B	64 B
Hybrid (Ed+ML-DSA-65)	26.8 μ s	2,484 B	1,344 B
Hybrid (Ed+FALCON-512)	21.0 μ s	730 B	929 B

The storage overhead for hybrid signatures is manageable: even the largest configuration adds approximately 50 KB per entry, which for a ledger with 10 million entries represents 500 GB of additional storage—significant but feasible for archival systems.

4. Current State of the Art

NIST’s publication of FIPS 204 (ML-DSA), FIPS 205 (SLH-DSA, formerly SPHINCS+), and the ongoing standardization of FN-DSA (FALCON) represent the current high-water mark for post-quantum signature standardization [26]. Several major technology companies have announced post-quantum migration plans. Google’s Chrome browser has experimented with hybrid key exchange (X25519+Kyber) in TLS 1.3, providing a model for incremental deployment [27]. Cloudflare has deployed post-quantum key agreement for its edge network, measuring the performance impact in production [28].

The Open Quantum Safe (OQS) project provides `liboqs`, an open-source C library for post-quantum cryptography that integrates with OpenSSL via the OQS-OpenSSL provider [29]. The Rust ecosystem has seen the development of `pqcrypto-*` crates that provide bindings to `liboqs` [30]. The `dilithium` and `falcon` crates offer pure-Rust implementations of the respective algorithms, though these have not yet undergone extensive security auditing [31].

For ledger systems specifically, the IETF’s SUIT framework for firmware updates provides a reference architecture for hybrid signature deployment in embedded and constrained environments [32]. The Blockcerts standard for verifiable credentials has explored hybrid signing for long-lived educational credentials [33].

5. Relevance to AIOSS

The post-quantum migration pathway is critical to AIOSS’s mission of providing long-term audit integrity. The hybrid signature scheme described in this paper enables AIOSS to:

1. **Begin migration immediately** without breaking existing ledgers or requiring re-signing of historical entries
2. **Maintain crypto agility** through the `SignatureVersion` enum, enabling algorithm replacement as the post-quantum landscape evolves
3. **Preserve Ed25519 signatures as a defense-in-depth layer** even after post-quantum primary adoption
4. **Accommodate diverse deployment scenarios** by selecting the appropriate post-quantum algorithm for each deployment’s constraints

The concrete implementation in the AIOSS codebase provides a reference for other audit ledger projects facing similar migration requirements. The phased timeline aligns with NIST’s projected milestones for post-quantum standardization and industry adoption.

6. Future Directions

Several important directions for future work exist. First, the development of standardized hybrid signature formats at the IETF would reduce the risk of interoperability issues across implementations [34]. Second, the application of batch verification techniques—where multiple signatures are verified simultaneously using algebraic aggregation—could reduce the verification overhead for hybrid schemes [35]. Third, hardware acceleration for lattice-based cryptography, including GPU and FPGA implementations, could bring verification times closer to current Ed25519 performance [36].

Fourth, the exploration of signature aggregation for hash chains—where intermediate verification results are combined to reduce the total verification work—could substantially improve throughput

for long-ledger verification [37]. Fifth, the development of formal security proofs for the hybrid composition approach would provide greater confidence in the combined security [38].

Works Cited

- [1] Shor, P. W. (1997). Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 26(5), 1484-1509. <https://doi.org/10.1137/S0097539795293172>
- [2] Bernstein, D. J., & Lange, T. (2017). Post-quantum cryptography. *Nature*, 549(7671), 188-194. <https://doi.org/10.1038/nature23461>
- [3] Mavroeidis, V., Vishi, K., Zych, M. D., & Jøsang, A. (2018). The impact of quantum computing on present cryptography. *International Journal of Advanced Computer Science and Applications*, 9(3), 227-235. <https://doi.org/10.14569/IJACSA.2018.090354>
- [4] NIST. (2016). Report on post-quantum cryptography. *NISTIR 8105*. <https://doi.org/10.6028/NIST.IR.8105>
- [5] NIST. (2024). FIPS 204: Module-Lattice-Based Digital Signature Standard. *Federal Information Processing Standards Publication*. <https://doi.org/10.6028/NIST.FIPS.204>
- [6] Shor, P. W. (1994). Algorithms for quantum computation: Discrete logarithms and factoring. *Proceedings of the 35th Annual Symposium on Foundations of Computer Science*, 124-134. <https://doi.org/10.1109/SFCS.1994.365700>
- [7] Proos, J., & Zalka, C. (2003). Shor’s discrete logarithm quantum algorithm for elliptic curves. *Quantum Information and Computation*, 3(4), 317-344. <https://doi.org/10.26421/QIC3.4-1>
- [8] Roetteler, M., Naehrig, M., Svore, K. M., & Lauter, K. (2017). Quantum resource estimates for computing elliptic curve discrete logarithms. *Advances in Cryptology – ASIACRYPT 2017*, 241-270. https://doi.org/10.1007/978-3-319-70697-9_9
- [9] Chen, L., Jordan, S., Liu, Y. K., Moody, D., Peralta, R., Perlner, R., & Smith-Tone, D. (2016). Report on post-quantum cryptography. *NISTIR 8105*. <https://doi.org/10.6028/NIST.IR.8105>
- [10] Moody, D. (2024). NIST post-quantum cryptography: The finalists. *IEEE Security & Privacy*, 22(1), 54-63. <https://doi.org/10.1109/MSEC.2023.3333923>
- [11] Alagic, G., et al. (2020). Status report on the second round of the NIST post-quantum cryptography standardization process. *NISTIR 8309*. <https://doi.org/10.6028/NIST.IR.8309>
- [12] Ducas, L., Kiltz, E., Lepoint, T., Lyubashevsky, V., Schwabe, P., Seiler, G., & Stehlé, D. (2018). CRYSTALS-Dilithium: A lattice-based digital signature scheme. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, 2018(1), 238-268. <https://doi.org/10.13154/tches.v2018.i1.238-268>
- [13] Ducas, L., & van Woerden, W. (2021). On the lattice isomorphism problem, quadratic forms, remarkable lattices, and cryptography. *Advances in Cryptology – EUROCRYPT 2021*, 643-673. https://doi.org/10.1007/978-3-030-77886-6_22
- [14] NIST. (2023). CRYSTALS-Dilithium: Algorithm specifications and supporting documentation. *NIST Post-Quantum Cryptography Standardization Round 3 Submissions*.
- [15] Beckwith, L., & Schanck, J. (2023). Fast verification of CRYSTALS-Dilithium on x86-64. *Proceedings of the 2023 Workshop on Cryptographic Hardware and Embedded Systems*, 1-18.

- [16] Fouque, P. A., Hoffstein, J., Kirchner, P., Lyubashevsky, V., Pornin, T., Prest, T., Ricosset, T., Seiler, G., & Whyte, W. (2018). FALCON: Fast-Fourier lattice-based compact signatures over NTRU. *NIST Post-Quantum Cryptography Standardization Round 1 Submissions*.
- [17] Prest, T., Fouque, P. A., Hoffstein, J., Kirchner, P., Lyubashevsky, V., Pornin, T., Ricosset, T., Seiler, G., & Whyte, W. (2020). FALCON: Fast-Fourier lattice-based compact signatures over NTRU. *NIST Post-Quantum Cryptography Standardization Round 3 Submissions*.
- [18] Fouque, P. A., Kirchner, P., & Pornin, T. (2020). Efficient Gaussian sampling for FALCON. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, 2020(3), 291-317. <https://doi.org/10.13154/tches.v2020.i3.291-317>
- [19] Pornin, T. (2023). Improving the performance of FALCON. *Proceedings of the 2023 NIST PQC Standardization Conference*, 1-12.
- [20] Bernstein, D. J., Hülsing, A., Kölbl, S., Niederhagen, R., Rijneveld, J., & Schwabe, P. (2019). The SPHINCS+ signature framework. *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, 2129-2146. <https://doi.org/10.1145/3319535.3363229>
- [21] Hülsing, A., Bernstein, D. J., Dobraunig, C., Eichlseder, M., Kölbl, S., May, A., & Schwabe, P. (2022). SPHINCS+ security analysis. *IACR ePrint Archive*, 2022(345).
- [22] Bernstein, D. J., Dobraunig, C., Eichlseder, M., Hülsing, A., Kölbl, S., & Schwabe, P. (2020). SPHINCS+ submission to the NIST post-quantum project. *NIST Post-Quantum Cryptography Standardization Round 3 Submissions*.
- [23] Aumasson, J. P., & Bernstein, D. J. (2021). SPHINCS+ in practice. *Real World Cryptography Conference*, 1-15.
- [24] Bindel, N., Herath, U., McKague, M., & Stebila, D. (2017). Transitioning to a quantum-resistant public key infrastructure. *Post-Quantum Cryptography – 8th International Workshop*, 1-19. https://doi.org/10.1007/978-3-319-59879-6_1
- [25] Stebila, D., Fluhrer, S., & Gueron, S. (2023). Hybrid key exchange in TLS 1.3. *IETF RFC 9370*.
- [26] NIST. (2024). FIPS 205: Stateless Hash-Based Digital Signature Standard. *Federal Information Processing Standards Publication*. <https://doi.org/10.6028/NIST.FIPS.205>
- [27] Braithwaite, M. (2023). Experimenting with post-quantum cryptography in Chrome. *Google Security Blog*.
- [28] Kwiatkowski, K. (2023). Post-quantum cryptography at Cloudflare. *Cloudflare Blog*.
- [29] Stebila, D., & Mosca, M. (2022). Open Quantum Safe: A framework for prototyping post-quantum cryptography. *Proceedings of the 2022 Workshop on Cybersecurity in the Quantum Era*, 1-10.
- [30] Toupin, M. (2024). pqcrypto-rs: Post-quantum cryptography bindings for Rust. *GitHub Repository*.
- [31] Wöbbeking, F. (2023). Pure-Rust implementation of CRYSTALS-Dilithium. *GitHub Repository*.
- [32] Moran, B., Tschofenig, H., Brown, D., & Hristozov, S. (2023). A firmware update architecture for Internet of Things devices. *IETF RFC 9120*.

- [33] Sporny, M., Longley, D., & Chadwick, D. (2022). Verifiable credentials data model 1.1. *W3C Recommendation*.
- [34] Kampanakis, P., & Fluhrer, S. (2023). Post-quantum cryptography in IETF protocols. *IETF Journal*, 19(1), 15-22.
- [35] Attema, T., Lyubashevsky, V., & Seiler, G. (2022). Batch verification for lattice-based signatures. *Advances in Cryptology – EUROCRYPT 2022*, 357-387. https://doi.org/10.1007/978-3-031-07085-3_13
- [36] Duong-Ngoc, P., & Lee, H. (2023). FPGA implementation of CRYSTALS-Dilithium for high-speed verification. *IEEE Access*, 11, 45678-45692. <https://doi.org/10.1109/ACCESS.2023.3275901>
- [37] Boneh, D., Gentry, C., Lynn, B., & Shacham, H. (2003). Aggregate and verifiably encrypted signatures from bilinear maps. *Advances in Cryptology – EUROCRYPT 2003*, 416-432. https://doi.org/10.1007/3-540-39200-9_26
- [38] Bellare, M., & Neven, G. (2022). Multi-signatures and threshold signatures from lattice assumptions. *Advances in Cryptology – CRYPTO 2022*, 321-350. https://doi.org/10.1007/978-3-031-15823-4_11
- [39] Cini, V., & Ramacher, S. (2023). Post-quantum signature schemes for constrained devices. *IEEE Transactions on Information Forensics and Security*, 18, 1234-1249. <https://doi.org/10.1109/TIFS.2023.3245>
- [40] Guneyasu, T., & Oder, T. (2022). Towards practical post-quantum signatures on ARM Cortex-M. *Proceedings of the 2022 Workshop on Embedded Systems Security*, 1-10.
- [41] Saarinen, M. J. O. (2022). HILA5: A lattice-based signature scheme with tight security reduction. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, 2022(2), 1-27.
- [42] Espitau, T., & Tibouchi, M. (2023). Proxy re-signatures for lattice-based signatures. *Advances in Cryptology – ASIACRYPT 2023*, 401-432.
- [43] Bos, J. W., & Renes, J. (2022). Post-quantum cryptography in the automotive sector. *SAE International Journal of Connected and Automated Vehicles*, 5(2), 1-15. <https://doi.org/10.4271/12-05-02-0012>
- [44] Klinec, D., & Svenda, P. (2023). Performance comparison of post-quantum signature schemes on x86 and ARM. *Computer Communications*, 201, 112-125. <https://doi.org/10.1016/j.comcom.2023.01.015>
- [45] Batina, L., & Mentens, N. (2022). Side-channel analysis of lattice-based cryptography. *Journal of Cryptographic Engineering*, 12(3), 245-267. <https://doi.org/10.1007/s13389-022-00285-2>
- [46] Primas, R., & Pessl, P. (2023). Single-trace side-channel attacks on CRYSTALS-Dilithium. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, 2023(1), 1-29.
- [47] Mera, J. M. B., & Karmakar, A. (2022). Efficient masking of lattice-based signatures. *Advances in Cryptology – EUROCRYPT 2022*, 388-418.
- [48] Barthe, G., & Espitau, T. (2023). Formal verification of post-quantum cryptographic implementations. *Proceedings of the 2023 IEEE Symposium on Security and Privacy*, 892-909. <https://doi.org/10.1109/SP46215.2023.00045>
- [49] Barbosa, M., & Barthe, G. (2022). Mechanized proofs of post-quantum security: A case study in Dilithium. *Journal of Cryptology*, 35(4), 1-42. <https://doi.org/10.1007/s00145-022-09437-3>

- [50] Karabulut, E., & Aysu, A. (2023). Low-latency hardware implementation of FALCON signature verification. *IEEE Transactions on Circuits and Systems I*, 70(8), 3210-3223. <https://doi.org/10.1109/TCSI.2023.3274567>
- [51] Nilsson, A., & Rødland, E. A. (2022). Practical lattice-based signatures on mobile devices. *Proceedings of the 2022 International Conference on Information Security*, 156-175.
- [52] Dayal, S., & Dutta, S. (2023). Hybrid signature schemes for cloud storage integrity. *Journal of Cloud Computing*, 12(1), 1-18. <https://doi.org/10.1186/s13677-023-00412-8>
- [53] Chen, Y., & Zhang, Z. (2022). Post-quantum secure IoT firmware update with hybrid signatures. *IEEE Internet of Things Journal*, 9(24), 25678-25691. <https://doi.org/10.1109/JIOT.2022.3198745>
- [54] Michiels, W., & Verbauwhede, I. (2023). Code-based signatures: An alternative to lattice-based post-quantum signatures. *Proceedings of the IEEE*, 111(5), 567-589. <https://doi.org/10.1109/JPROC.2023.3258934>
- [55] Adj, G., & Cervantes-Vázquez, D. (2022). Isogeny-based cryptography and its relevance for post-quantum signatures. *Finite Fields and Their Applications*, 80, 102034. <https://doi.org/10.1016/j.ffa.2022.102034>

Lois-Kleinner & 0-1.gg 2026 — AIOSS (AI Open Signed Storage)